УДК 004.49

СТРУКТУРНА ПОДІБНІСТЬ МЕТАМОРФНОГО ШКІДЛИВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

В. Ю. Кожокар[1, a], І. В. Стьопочкіна[1]

[1] *Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Фізико-технічний інститут*

Анотація

Робота присвячена знаходження статистичних властивостей у метаморфних вірусах. Показано яким чином можуть бути ідентифікованими обфусковане шкідливе програмне забезпечення, віруси.

Ключові слова: шкідливе програмне забезпечення, техніки метаморфування, структурна подібність

Вступ

Сучасні шкідливі програми набувають здатності самостійно модифікувати себе, для того, щоби приховувати свою сутність від антивірусних засобів, які працюють за сигнатурним принципом. Розробники шкідливих програм створюють спеціальні метаморфні модулі, які здатні змінювати зовнішній вигляд шкідливої програми без зміни її функціоналу. Отже, засоби аналізу відповідних програм стикаються із новою сигнатурою для насправді існуючого вірусу і не можуть його розпізнати [1]. У найкращому випадку це утруднює класифікацію та кластеризацію зразка шкідливого ПЗ, а в найгіршому — перешкоджає впізнаванню зразка як небезпечного. Засоби поведінкового аналізу в цих випадках теж можуть виявитись неефективними, оскільки ряд зразків шкідливого ПЗ вміє розпізнавати віртуалізацію та змінює свою поведінку. Таким чином, дослідження статичних характеристик шкідливого ПЗ та встановлення певних спільних закономірностей, притаманних нащадкам одного й того самого зразка, залишається актуальною задачею.

1. Існуючі техніки метаморфування

Основні техніки, які використовуються метаморфними вірусами, полягають у застосуванні наступних прийомів (фактично, це типові прийоми обфускації):

- заміна одних конструкції мови асемблер іншими, з іншим бінарним представленням, однак із тим самим функціоналом,
- введення додаткових блоків, які не змінюють функціонал програми, до структури програми,
- перестановки вершин графу виконання програми, які не змінюють функціонал.

[a] vladislav.kozhokar@gmail.com

2. Архітектура та принцип еволюції метаморфного вірусу

Зазвичай метаморфний вірус має містити такі основні модулі: дизасемблер, оптимізатор/стискувач опкодів (`shrinker`), розширювач опкодів (`expander`), перестановщик (`swapper`), переміщувач (`relocator`), засмічувач (`garbager`) та прибиральник. Внутрішній дизассемблер виконує зворотні перетворення коду. Shrinker стискає, оптимізує дві та більше інструкцій в одну. Розширювач, навпаки, розширює одну інструкцію до більш ніж однієї. Перестановшик міняє місцями дві чи більше команд. `Relocator` перераховує всі відносні посилання (стрибки, виклики і покажчики). `Garbager` встановлює одну чи більше порожніх інструкції між реальним кодом (це можуть бути nop, або комбінації інструкцій). Прибиральник, навпаки, прибирає засмічуючий код, вставлений засмічувачем.

Еволюція метаморфного вірусу відбувається в наступні етапи [2]:

1) Вірусний код дизасемблюється в проміжну форму, яка не залежить від апаратної платформи (процесора), на якій виконується код. Це робить можливим створення коду для різних операційних систем або навіть різних процесорів.
2) Проміжна форма скорочується шляхом видалення зайвих та невикористовуваних інструкцій. Ці інструкції були додані в попередніх реплікаціях, щоб заважати дизасемблюванню сторонніми особами.
3) Виконуються перестановки підпрограм чи блоків коду, пов'язуючи їх з інструкціями переходу.
4) Код знов розширюється шляхом додавання надлишкових і невикористовуваних інструкцій.
5) З проміжної компонується кінцева форма, яка буде додана до заражених файлів.

Табл. 1. Еволюція оригіналу: опкоди Ассемблер

Інструкції Clibo	Оригінал	Модифік.1	Модифік.2	Модифік.3
Mov	631	513	512	512
Add	138	131	131	131
Dec	43	43	43	43
Push	50	48	49	49
Cmp	85	69	69	69
Ret	66	62	62	62
Lea	85	79	79	79
Test	76	55	57	58
And	60	56	56	56
Pop	70	66	67	67
Xor	44	32	32	32
Sub	449	449	449	449

3. Характеристики та методи аналізу метаморфних зразків

В якості характеристик, за якими можливо здійснювати аналіз метаморфних вірусів, можна вказати наступні:

1) тип опкодів;
2) кількість опкодів певного виду;
3) наявність підозрілих операцій (чи опкодів, які зазвичай рідко зустрічаються)
4) характерні зміни у структурі PE-заголовку (ці характеристики скоріше використовувані для загального визначення шкідливого ПЗ);
5) наявність спільних закономірностей в структурі бінарного файла.

4. Результати експерименту

В даній роботі здіснено порівняння кількості опкодів (мова асемблер) для вихідного зразка та нащадків, одержаних для нього за допомогою технік метаморфування. В якості вихідного зразка було обрано віруси із бази vxheaven, до яких застосовано метаморфний модуль metame. Одержано табличні дані, типовий приклад наведено в таблиці 1.

З таблиці видно, що кількість опкодів відповідного виду змінюється у нащадків незначно. Різниця між оригінальним зразком та першим нащадком є вищою, однак теж відрізняється менше, ніж на порядок. Показники кількості типів різних опкодів, відповідно, для різних зразків, суттєво відрізняються. Для різниці між гістограмами опкодів можна використовувати метрику Мінковського, де координатами векторів є кількість опкодів кожного виду.

При аналізі кількості змін інструкцій встановлено, що в першого потомка порівняно з батьком змінено (в середньому) 9 тис. інструкцій, між першим та другим відмінність становить 5 тисяч інструкцій, і надалі середня кількість змін становить 4-5 тис. інструкцій.

Висновки

Незважаючи на складну природу метаморфного перетворення, існує можливість встановити відповідність між зразками одного й того ж самого оригіналу. В числі методів, які можуть бути використані, є статистичний аналіз кількості та типів опкодів. Експериментально показано, що різниці в кількостях опкодів одного виду нащадків є незначними, і середні кількості змін в інструкціях між нащадками одного оригіналу є в середньому однаковими. Ці результати можуть бути використані для розв'язання задачі кластеризації зразків шкідливого ПЗ, а в деяких випадках, коли є можливість встановити відповідність між оригіналом та першим нащадком — й для задачі класифікації.

Перелік використаних джерел

1. Sathyanarayan V.S., Kohli P. Signature Generation and Detection of Malware Families. — 2008. — P. 336–349. — Access mode: `https://vxheaven.org/lib/pdf/Signature...Families.pdf`.
2. E. Konstamtinou. Metamorphic Virus: Analysis and Detection / Technical Report RHUL-MA-2008-02. — 2008. — P. 88 c. — Access mode: `https://www.ma.rhul.ac.uk/static/techrep/2008/RHUL-MA-2008-02.pdf`.